

# C++ Abstract Class

郑华

2017 年 2 月 4 日

## 1 概要

纯虚函数是一种特殊的虚函数，它的一般格式如下：

```
class <类名>
{
    virtual <类型><函数名>(<参数表>)=0;
    ...
};
```

在许多情况下，在基类中不能对虚函数给出有意义有实现，而把它说明为纯虚函数，它的实现留给该基类的派生类去做。这就是纯虚函数的作用。下面给出一个纯虚函数的例子。

```
class point
{
public:
    point(int i=0, int j=0) { x0=i; y0=j; }
    virtual void set() = 0;
    virtual void draw() = 0;

protected:
    int x0, y0;
};

class line : public point
{
public:
    line(int i=0, int j=0, int m=0, int n=0):point(i, j)
    {
        x1=m; y1=n;
    }
    void set() { cout<<"line::set() called."; }
    void draw() { cout<<"line::draw() called."; }

protected:
    int x1, y1;
};

class ellipse : public point
{
public:
    ellipse(int i=0, int j=0, int p=0, int q=0):point(i, j)
    {
        x2=p; y2=q;
    }
    void set() { cout<<"ellipse::set() called."; }
    void draw() { cout<<"ellipse::draw() called."; }
```

```

protected:
    int x2, y2;
};

void drawobj(point *p)
{
    p->draw();
}

void setobj(point *p)
{
    p->set();
}

void main()
{
    line *lineobj = new line;
    ellipse *elliobj = new ellipse;
    drawobj(lineobj);
    drawobj(elliobj);
    setobj(lineobj);
    setobj(elliobj);
    cout<<"Redraw the object...";
    drawobj(lineobj);
    drawobj(elliobj);
}

```

执行结果:

```

line::draw() called.
ellipse::draw() called.
line::set() called.
ellipse::set() called.
Redraw the object...
line::draw() called.
ellipse::draw() called.

```

## 2 抽象类

带有纯虚函数的类称为抽象类。抽象类是一种特殊的类，它是为了抽象和设计的目的而建立的，它处于继承层次结构的较上层。抽象类是不能定义对象的，在实际中为了强调一个类是抽象类，可将该类的构造函数说明为保护的访问控制权限。

抽象类的主要作用是将有关的组织在一个继承层次结构中，由它来为它们提供一个公共的根，相关的子类是从这个根派生出来的。

抽象类刻画了一组子类的操作接口的通用语义，这些语义也传给子类。一般而言，抽象类只描述这组子类共同的操作接口，而完整的实现留给子类。

抽象类只能作为基类来使用，其纯虚函数的实现由派生类给出。如果派生类没有重新定义纯虚函数，而派生类只是继承基类的纯虚函数，则这个派生类仍然还是一个抽象类。如果派生类中给出了基类纯虚函数的实现，则该派生类就不再是抽象类了，它是一个可以建立对象的具体类了。